Undergraduate Teaching Assistant Written Feedback: Coding Synthesis, Analysis and Comparison

THESIS

Presented in Partial Fulfillment of the Requirements for the Honors Research Distinction in the Undergraduate School of The Ohio State University

By

Connor B. Jenkins

Undergraduate Program in Electrical and Computer Engineering

The Ohio State University

2021

Undergraduate Research Committee:

Professor Krista Kecskemety, Advisor

Professor Annie Abell, Committee Member

Abstract

Technical communication, any form of communication that must effectively convey specialized knowledge such as lab reports, product proposals, etc. is a large part of many engineers' lives. It is important for engineering students to learn technical communication skills because of the influence it may have on their future careers. At Ohio State, technical communication skills are taught in the Fundamentals of Engineering courses through technical writing assignments such as lab reports. Students can track their mastery of technical communication using their scores on these assignments and the written feedback left on their work by Undergraduate Teaching Assistants (UTAs). Despite the positive impact written feedback has on student learning, the quality of and experiences that inform UTA written feedback are largely unknown. This study aims to be the first step in improving UTA written feedback methods.

A group of UTAs, Graduate Teaching Assistants (GTAs), and faculty were given a student writing sample to score and leave written feedback on. Their comments were broken into individual ideas and then coded using two different coding methods, one focusing on the content of the ideas, and the other focusing on the purpose. The two coding methods were synthesized from literature that discussed categorizing feedback on student work. From these results, trends and observations comparing the UTAs and Experts (GTAs and Faculty) and described alongside observations of UTAs alone. In general, UTAs and Experts do not share common written feedback methods. Future work will explore the experiences that inform UTA written feedback methods through a focus group with UTA participants from this study.

Table of Contents

Background & Motivation

Technical communication is any form of communication that includes specialized technical information such as scientific reports, technical manuals, presentations, or communications with colleagues [1]. In the workplace, engineers often spend most of their time communicating technical information [1], [2]. Because of this, graduate engineers have frequently expressed the importance of strong technical communication skills in their careers [2], [3], [4]. The importance of technical communication skills is also supported by the Accreditation Board of Engineering Technology (ABET) student outcome, "an ability to communicate effectively with a range of audiences" [5], that is required to be met by all accredited engineering programs.

Technical communication skills are taught to engineering students through assignments that include lab reports, technical presentations, and/or other technical documents. Two ways students can be given insight on their progress towards mastering technical communication is through scores on assignments, or through feedback left by a reviewer. Multiple studies have shown that student learning is enhanced through the use of feedback on assignments [6], [7], [8], so this method can be particularly constructive for students. The effectiveness of feedback can be influenced by many factors such as who leaves the feedback, time between assignment completion and feedback review, the content of the feedback, and the amount of feedback received [6], [7], [8], [9], [10]. To maximize student mastery of technical communication skills, these factors should all be considered when reviewing student technical communication assignments.

At Ohio State, the Fundamentals of Engineering Program (FEP) courses are one way engineering students learn technical communication skills. The FEP courses are taught by a teaching team consisting of one faculty member, one graduate teaching assistant (GTA) and three or more undergraduate teaching assistants (UTAs). The faculty deliver most of the content during lectures, while the GTAs lead labs, and the UTAs assist students as needed. This format is consistent between the two primary FEP course tracks: Fundamentals of Engineering – Standard (FE) and Fundamentals of Engineering – Honors (FEH). Both course tracks teach technical communication skills to the students through lecture and lab instruction, and reinforce the ideas through assigning presentations, lab reports, and other technical documents [11], [12].

In addition to assisting students during class, the FEP UTAs are responsible for grading most course assignments, providing feedback on student work, and answering student questions related to the course. To ensure UTAs have all required skills necessary to handle the job responsibilities, they undergo frequent training throughout the semester. The training is constructed by FEP and Engineering Technical Communications (ETC) faculty and managed by a small group of GTAs and UTAs. Failure to complete the necessary training can impact re-hiring decisions and could potentially lead to immediate termination [13]. The primary training on written feedback practices for UTAs is a one-time grading training at an orientation conducted before classes start in the autumn semester where UTAs receive feedback on their scoring and written feedback of a large technical writing assignment (e.g., a lab report).

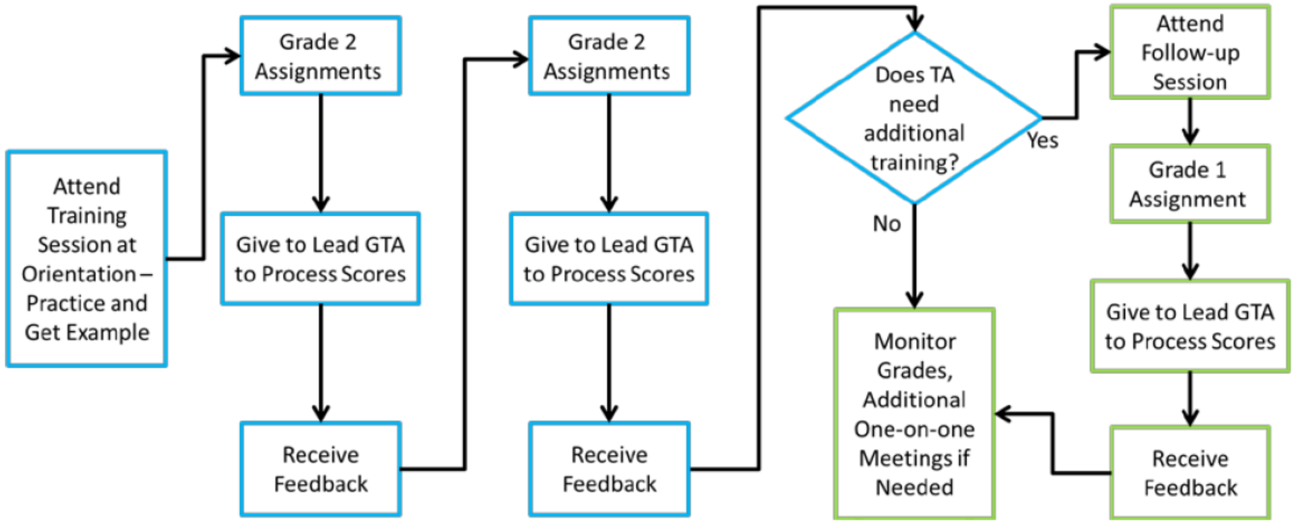

Figure 1: TA Training and calibration process examined by Kecskemety et al.

The last significant study examining the teaching assistant (TA) training methodology used the training procedure shown in Figure 1 [14]. This procedure was primarily concerned with the scoring of student assignments and only addressed written feedback through some discussion and examples during the original training session [14]. The discrepancies between this procedure and the current training methodology are due to time constraints and variations in the personnel responsible for the training. Current training practices are limited in addressing UTA written feedback in a similar fashion.

Despite the importance of quality written feedback on student achievement, the quality of UTA written feedback was not previously analyzed by Kecskemety et al. While the current training does incorporate some discussion of written feedback methods, the effectiveness of this training, and the influence it may have on UTA feedback are unknown. This research aims to be the first step in improving UTA written feedback methods by answering the following research question: **How does the written feedback UTAs in FEP courses provide on technical writing**

assignments compare to known best practices for written feedback, and feedback given by engineering content experts and technical communication experts?

Data Collection

Participants for this research study were recruited from the following groups: ETC and FEP faculty, FEP GTAs, and FEP UTAs. A total of thirteen participants were recruited. Each participant was given a number which will be used to identify them throughout the remainder of this paper. The participant information is summarized in Table 1. Participants 3 and 9 were unable to complete the required task and were excluded from the table, and from all further discussion.

*Table 1:*Participant Summary

Number	Role	Course
1	UTA	FEH
2	UTA	FEH
4	UTA	FE
5	UTA	FE
6	UTA	FE
7	UTA	FE
8	UTA	FE
10	GTA	FEH
11	FEP Faculty	FE
13	ETC Faculty	-

Each participant was asked to grade and leave written feedback on a writing sample as if it were a real submission by a student. To ensure the sample was as realistic as possible, an actual student lab report was used. Care was taken to ensure all identifying information was removed from the sample. Additionally, no participant had seen the report prior to participating in the study to help further protect the student's identity. The experiment that the report discussed is conducted in both FEP courses, however a full report is only submitted in the FEH courses. The submission for the FE courses has varied between several different technical documents depending on the year. The same prompt and directions used by the student were given to the participants to account for any difference in directions between the FEP courses. The prompt also served to give all participants, especially the ETC faculty who may not be familiar with the lab, the same context when approaching the sample.

Because the two FEP course tracks have different assignments for this experiment the rubric used in both courses for the lab assignment are also different. A rubric for the sample was synthesized by comparing a lab report rubric from FE and a lab report rubric from FEH. The goal of the synthesis was to ensure that no participant was more familiar with the rubric than any other participant. By using an unfamiliar rubric, all participants had to take care when going through the rubric, rather than relying on familiarity to move quickly through the process.

The sample was hosted on Canvas Learning Management Software, the same software used by the FEP courses to grade student work. Directions on how to use the software were also provided to the participants, including how each method of commenting can be accessed. Examples of these directions are in Figure 2.

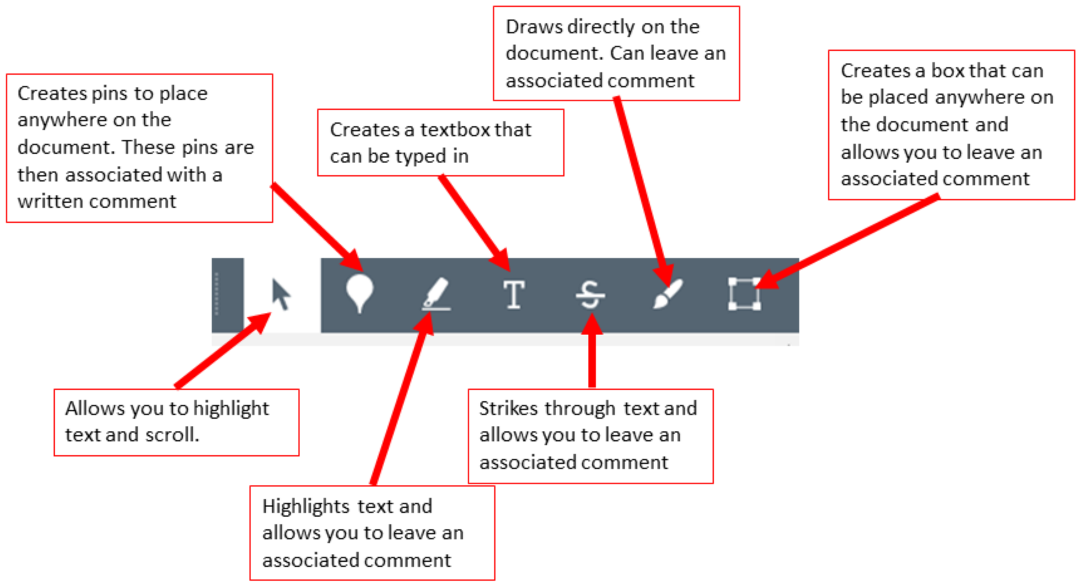

Figure 2: Commenting tool directions given to participants

Participants were encouraged to work through the sample at their own pace and on their own time to mimic UTA working environments with student writing. Under normal circumstances, UTAs are expected to return several graded reports within a week of submission [13]. This time constraint was not enforced on the participants to allow for more flexibility in participation. Participants were instead given multiple weeks to finish the task. Participants were then asked to mark their work as complete once they had finished the task. Once the deadline had passed, all participants who had not marked their work as complete were contacted to determine if the task was incomplete or if they had forgotten to mark the work as complete.

Coding Scheme Synthesis and Method

Three areas of interest emerged when examining the written feedback left by the participants:

1. Method of feedback – What tool was used to leave the comment?
2. Quantity – How many comments were left?
3. Content – What do they comments say?

While method and quantity can be analyzed numerically, the content of each comment is more abstract. Analyzing these comments falls under the broad definition of content analysis: a systematic analysis of message characteristics [15]. One method of conducting this analysis is through the creation and utilization of a coding scheme [15]. A coding scheme consists of a codebook designed to be as clear and complete as possible and a coding form for the researchers to track the codes throughout the content of interest [15]. Instead of creating a codebook from scratch, existing categorization methods for feedback were examined to determine their viability as codebooks. While many categorization methods for feedback exist [6], [7] , [16], [17], [18], [19], [20], [21], [22], each have limitations and cannot cover all possible comments.

It is important to note that while the goal of using a coding scheme for this study was to code only written feedback data, all subsequent categorization methods discussed were created to be applied to all manner of feedback. This includes scores, verbal feedback, computer-based feedback, etc. Since not all categories in each method were meant to be used for written feedback, individual methods may not be fully capable of coding written feedback in detail. To mitigate the limitations and maximize the coverage of the codes for written feedback, multiple categorization methods were examined. After an initial examination of existing categorization methods, two groups were created to organize the classification methods by their classification mechanism. The categories are content classification, which focuses on the information shared in the feedback, and functional classification, which focuses on the purpose of the feedback.

Content Classification

Content classification methods focus on the explicit content of the feedback. They answer the question: What information does the feedback contain? The basis for many content-based categorization methods comes from the definition of feedback expressed by Kulhavy et al [16]:

$$Feedback = Verification + Elaboration \tag{1}$$

Where the verification portion expresses whether something is correct or incorrect, and elaboration is any additional information expressed in the feedback. Kulhavy et al also expressed a categorization method shown in Table 2. In addition to the three categories, Kulhavy et al differentiated within each code using two additional characteristics: Form and Load. Form was related to changing the structure of the information presented. For the feedback left by the participants, the form would be related to both the syntax of the comment and the Canvas LMS tool used. The complexity of the feedback was characterized using Load; however, no measure was recommended for use. Because of the generality of the categories, these have been built upon by other researchers to create more specific coding books.

Table 2: Categorization method shared by Kulhavy et al

Code	Definition
Task-Specific	Information about the task (e.g., restating the question)
Instruction Based	Information from relevant previous instruction
Extra-Instructional	Addition of new information

Table 3 shares a categorization method [17] that was synthesized using categorization methods shared in [16], [23], [24], [25], [26], [27] by Mason et al. Here, an item refers to a single instructional task demand [16], such as a multiple-choice question. All 8 categories rely on the definition of feedback shared in Equation 1, with all categories include individual verification of each item besides “No Feedback”. The primary difference between the 7 categories that include item verification is the elaboration aspect of the feedback. For example, the difference between topic and response contingent coded feedback is the relationship between the feedback and the student response.

Table 3: Content classification categories synthesized by Mason et al.

Code	Definition
No Feedback	Only the number or proportion of correct responses
Knowledge of Response	Individual verification of response for each question
Answer until Correct	Only item verification, and forces student to remain on the same item until correct
Knowledge of Correct Response	Individual item verification and correct answer
Topic Contingent	Item verification and general information about item's general topic (e.g., given extensive information, but students must find it themselves)
Response Contingent	Item verification and item specific elaboration (e.g., explanation as to why the incorrect answer was wrong)
Bug Related	Item verification and addressing specific errors that are placed in a list of "common student errors" for student review
Attribute Isolation	Item verification and highlights the key components of target concepts

While these categories improve on the differentiation of ideas from Kulhavy et al, they struggle to be applied to written feedback in this study. One category that illustrates this difficulty is “Answer until Correct”. In the context of writing, this would manifest as a reviewer writing with the student and then stopping the student if they made a mistake. Because the writing sample was complete when the participants were writing feedback, they could not generate feedback coded as “Answer until Correct”. This category is more applicable to a multiple-choice style assessment where there is a definitive correct answer and feedback can be returned immediately.

Shute et al created a categorization method that built upon the work of both Kulhavy et al and Mason et al with the goal of organizing the complexity of the elaboration component of the feedback is listed in Table 4 [7]. Shute et al separated the verification and elaboration components of feedback. The first 5 categories relate to the verification component and the last 6 relate to the elaboration component of feedback.

Table 4: Categories organized primarily by complexity of elaboration from Shute et al

Code	Definition
No Feedback	No indication of the correctnesess of a response
Verification/Knowledge of Results	Presents correctness of student response
(Knowledge of) Correct Response	Only gives the correct answer to a problem
Try-Again/Repeat Until Correct	If incorrect, student receives one or more additional attempts to answer correctly
Error-Flagging/Location of Mistakes	Highlights errors without giving correct answer
Elaborated	General term to refer to next 6 codes. Provides an explanation about why a specific response is correct
Attribute Isolation	Highlights the key components of target concepts
Topic Contingent	Gives information relating to target topic
Response Contingent	Focuses on the specific response, such as describing why an answer is wrong
Hints/Cues/Prompts	Guiding in the direction of the correct answer without explicitly giving it.
Bugs/Misconceptions	Provides information about the student's specific errors or misconceptions
Informative Tutoring	Presents verification feedback, error-flagging, and strategic hints on how to proceed

Separating verification and elaboration allows for greater distinction between ideas but some categories are still general and can be made more specific. For example, “Topic Contingent” does not differentiate between supplying the student with new information or re-presenting information that has already been distributed. Shute et al also has a similar issue with the category “Try-Again” as previously discussed with the category “Answer until Correct” presented by Mason et al.

Table 5: Categories differentiated by information feedback addresses by Narciss

Code	Definition
Knowledge of Performance (KP)	Score/Percentage correct
Knowledge of Response (KR)	Informs if response is correct or incorrect
Knowledge of the Correct Result (KCR)	Gives the correct response
Knowledge about Task Constraints (KTC)	Hints/explanations on type of task Hints/explanations on task-processing rules Hints/explanations on subtasks Hints/explanations on task requirements
Knowledge about Concepts (KC)	Hints/explanations on technical terms Examples illustrating the concept Hints/explanations on the conceptual context Hints/explanations on concept attributes Attribute-isolation examples
Knowledge about Mistakes (KM)	Number of mistakes Location of mistakes Hints/explanations on types of errors Hints/explanations on sources of errors
Knowledge about How to Proceed (KH)	Bug-related hints for error correction Hints/explanations on task-specific strategies Hints/explanations on task-processing steps Guiding questions Worked-out examples
Knowledge about Metacognition (KMC)	Hints/explanations on metacognitive strategies Metacognitive guiding questions

Narciss created a categorization method that built upon Mason et al’s work focused on categorizing the elaboration portion of the feedback by the context of the information the feedback addressed [18]. Table 5 shows the 8 total categories, with the last 5 focused on the elaboration portion of feedback. By focusing on the context instead of just the raw information, categories present Mason et al and Shute et al are distributed throughout several categories in Table 5. E.g., “Hints/Cues/Prompts” from Table 4 could be coded in any of the 5 elaboration categories in Table 5 depending on the context of the hint. The focus on information also removes procedural issues that arise when looking at written feedback, specifically the "Try-Again/Answer until Correct" error discussed previously.

Table 6: Summary of content classification methods *[7], [17], [18]*

Code	Source	Definition
Knowledge of Performance/ No Feedback	Narciss, Mason, Shute	Proportion of correct responses
Knowledge of Response/Verification	Narciss, Mason, Shute	Individual item verification
Knowledge of the correct result	Narciss, Mason, Shute	Item verification and given correct answer
Topic Contingent	Mason, Shute	Item verification plus information about where the answer is found
Response Contingent	Mason, Shute	Extra-instructional feedback that focuses on specific response
Bug Related	Mason, Shute	Item verification plus specific errors
Attribute Isolation	Mason, Shute	Item verification plus highlighting key concpets
Answer until Correct	Mason, Shute	Remain on the same question until correct
Knowledge about task constraints	Narciss	Information about type of task, task requirements, etc.
Knowledge about concepts	Narciss	Information about concept attributes, context, technical terms, etc.
Knowledge about how to proceed	Narciss	Error correction, task-specific strategies or steps, guiding questions or examples
Knowledge about metacognition	Narciss	Metacognitive strategies or guiding questions
Knowledge about Mistakes/Error-Flagging	Narciss,Shute	Highlights errors in solution without giving correct answer
Elaborated	Shute	Provides explanation about why a specific response is correct
Hints/Cues/Prompts	Shute	Guides reader in the right direction while avoiding explicitly presenting answer
Informative Tutoring	Shute	Presents verification, error-flagging, and hints on how to proceed

All three categorization methods discussed share 3 categories, although they differ in name. Similarly, the schemes presented in Mason et al and Shute et al share an additional 5 categories. Table 6 consolidates all three categorization methods and lists the common categories between sources. The first three categories handle the verification element of feedback for all three categorization methods. Because of this shared idea of necessary verification differentiation, these three were included in the final content classification method.

Despite Mason et al and Shute et al having the most common categories between the schemes, they proved difficult to use with comments on a writing sample. General definitions in categories such as “Topic and Response Contingent” led to confusion when coding ideas. Instead, the categories from Narciss’ work listed in Table 5 were used to categories the elaboration component of feedback in the final content classification method because they can differentiate between similar ideas, do not include categories that are inapplicable to feedback on writing, and include specific definitions.

Functional Classification

While content classification covers the explicit content written in the feedback, functional classification methods are concerned with the reason the feedback was left. Because they rely on the purpose of the feedback, rather than explicit content, they are more difficult for the coder to use. The coder must determine the intent of the original reviewer using only the comment and any surrounding information.

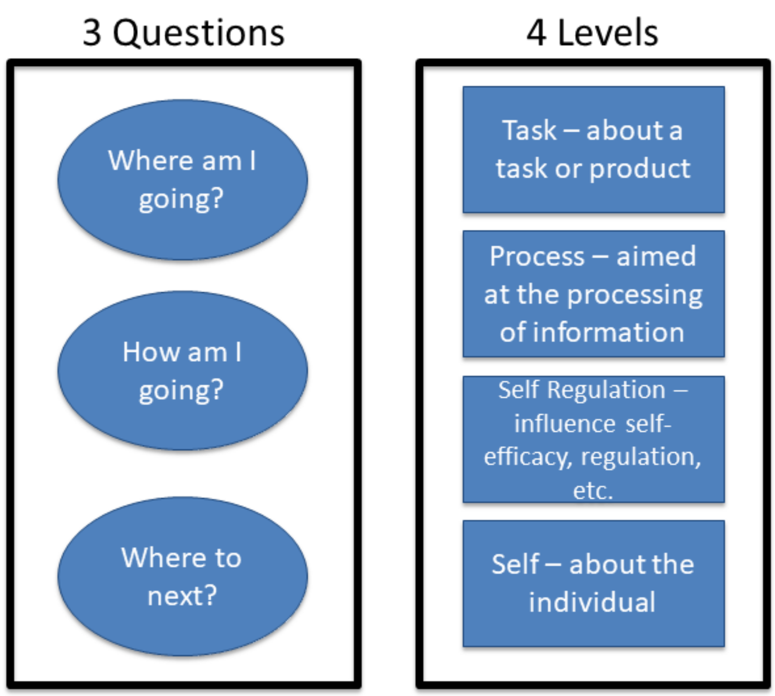

Figure 3: Questions, levels, and definitions for the categorization method by Hattie et al

Hattie et al presented feedback as a way to reduce discrepancies between student knowledge levels and the desired knowledge level [6]. Feedback is categorized by the “level” at which it answers one of three questions [6]. These levels and questions are shared in Figure 3. While this scheme offers 12 unique categories by which to identify feedback with, their definitions are not well defined.

Table 7: Categories grouped by framework *[18]*

Group	Code	Definition
Cognitive	Informative	Give number, location, type of or reason for error
	Completion	Provides information with missing knowledge
	Corrective	Provides information that corrects incorrect elements
	Differentiation	Gives clarity to imprecise content
	Restructing	Corrects connections between elements that are incorrect
Metacognitive	Informative	Provides information about metacognitive strategies
	Specification	Provides criteria for monitoring goals or where conditions for strategies are specified
	Corrective	Corrects erroneous metacognitive strategies
	Guiding	Encouraged to generate own monitoring or evaluation of solution strategies
Motivational	Incentive	Renders the results of task processing visible
	Task	Contribute information for overcoming task difficulties
	Self-Efficacy Enhancing	Provides information that makes it possible to master tasks successfully regardless of mistakes made
	Reattribution	Information that contributes to mastery expereicnes that can be linked to personal causes

Table 7 shows three separate frameworks by which feedback can be categorized, alongside corresponding categories for each framework [18]. Utilizing any of the coding requires a strong understanding of the theoretical framework that the categories were built upon. While frameworks provide a structure to analyze information through, the coding process would be complicated by misunderstandings of both the categories themselves, and the framework that backs the categories. Another issue with the three categorization methods presented in Table 7 is the lack of specificity in the category definitions, as well as the gaps that may appear when coding due to the small number of categories and framework specific definitions. For example, it would be difficult to code a simple typo correction if only the Metacognitive categories were available.

Next, four classification methods created by Cusella [19], Sales [20], Wager et al [21], and Butler et al [22] were compiled into groups by Narciss [18]. These methods and groupings are shown in Table 8. All four sources had at least one category in each group created by Narciss, but many sources also had unique categories within each group. Using the groups created by Narciss would require synthesizing the categories within each group to form a new category definition. This synthesis would require combining definitions of terms such as “Advising” and “Indicating” that do not have the same meaning. Instead, the categories common to more than one source were collected in Table 9. While each source had a slightly different definition for each function, the definitions were similar which simplified the synthesis of a new definition.

Table 8: Functional classification methods and groups *[18], [19], [20], [21], [22]*

Groups made by Narciss	Cusella	Sales	Wager and Morty	Butler and Winne
Acknowledge/Reinforce Functions			Confirming	Confirming
	Reinforcing			
		Assessing	Assessing	
Informing Functions	Informing	Informing	Informing	Informing
Steering/Guiding Functions		Advising		
		Guiding		
				Making suggestions
	Indicating		Indicating	Indicating
Regulatory/Correcting Functions	Regulating	Regulating	Correcting	Correcting
Motivational Functions	Motivating	Motivating	Motivating	
		Stimulating		
Instructing Functions	Instructing	Instructing	Instructing	
				Completing knowledge
				Differentiating
				Restructuing

The categories in Table 9 do not rely on strong background knowledge in theoretical frameworks, nor do they require multiple decisions per coding decision. The goal for this reduction in complexity was to increase the clarity of the categories while not losing information from the literature. These categories also show a high level of agreement in the literature, with 5 out of the 7 categories appearing in 3 or more separate sources. Because these functions reduce the complexity of the coding process and are based in the literature, they were selected to be used in the functional classification method.

Table 9: Summary of Functional Classification methods *[19], [20], [21], [22]*

Functions	Source	Definitions
Confirming	Wager, Butler	Tells students that their understanding of the material is correct
Assessing	Sales, Wager	Evaluates student performance
Informing	Cusella, Sales, Wager, Butler	Add information that the students may lack
Indicating	Cusella, Wager, Butler	Show area of interest for student to re-examine themselves
Correcting/Regulating	Cusella, Sales, Wager, Butler	Provide information meant to replace inaccurate student knowledge or misconception
Motivating	Cusella, Sales, Wager	Encourages/facilitates student desire to learn/continue effort
Instructing	Cusella, Sales, Wager, Butler	Tweak a mostly correct understanding, differentiate between similar concepts, specifying conditions for rules, provide supplemental information meant to enhance understanding further, etc.

The final codebooks for the functional and content classification methods are shown in Table 10 and Table 11 respectively. Both tables contain the codes and the definitions that will be used throughout the remainder of this study. Acronyms are also provided for the content classification codebook to aide in referencing the codes.

Table 10: Functional Classification Codebook

Code	Definition
Confirming	Tells students that their understanding of the material is correct
Assessing	Evaluates student performance
Informing	Add information that the students may lack
Indicating	Show area of interest for student to re-examine themselves
Correcting/Regulating	Provide information meant to replace inaccurate student knowledge or misconception
Motivating	Encourages/facilitates student desire to learn/continue effort
Instructing	Tweak a mostly correct understanding, differentiate between similar concepts, specifying conditions for rules, provide supplemental information meant to enhance understanding further, etc.

Table 11: Content Classification Codebook

Code	Definition
Knowledge of Performance (KP)	Score/Percentage correct
Knowledge of Response (KR)	Informs if response is correct or incorrect
Knowledge of the Correct Result (KCR)	Gives the correct response
Knowledge about Task Constraints (KTC)	Hints/explanations on type of task Hints/explanations on task-processing rules Hints/explanations on subtasks Hints/explanations on task requirements
Knowledge about Concepts (KC)	Hints/explanations on technical terms Examples illustrating the concept Hints/explanations on the conceptual context Hints/explanations on concept attributes Attribute-isolation examples
Knowledge about Mistakes (KM)	Number of mistakes Location of mistakes Hints/explanations on types of errors Hints/explanations on sources of errors
Knowledge about How to Proceed (KH)	Bug-related hints for error correction Hints/explanations on task-specific strategies Hints/explanations on task-processing steps Guiding questions Worked-out examples
Knowledge about Metacognition (KMC)	Hints/explanations on metacognitive strategies Metacognitive guiding questions

Coding Method

Before coding could occur, the comments were broken up by individual ideas. Typically, this meant that each sentence from a comment became one idea. Some comments were not written out in full sentences and other comments were lists of many related thoughts. In these cases, the phrases were split to separate out each idea.

To capture both the content and the function of each idea in the coding process, both classification methods were used such that each idea received both a content and functional classification. Because the coding required decisions to be made by individuals and was influenced by personal experience, the coding was completed by two researchers. The researchers first independently coded one participant's ideas using both methods. Then each classification that was not initially agreed upon was discussed until a consensus was met. This was repeated for another participant. Next, half of the remaining ideas were coded by the researchers, and then discussed. Finally, the remaining ideas were coded, and a final discussion took place. This method is summarized in Figure 4. The goal of this method was to ensure 100% agreement between researchers for each idea while also refining the researchers' understanding of the codes in the context of written feedback on technical writing.

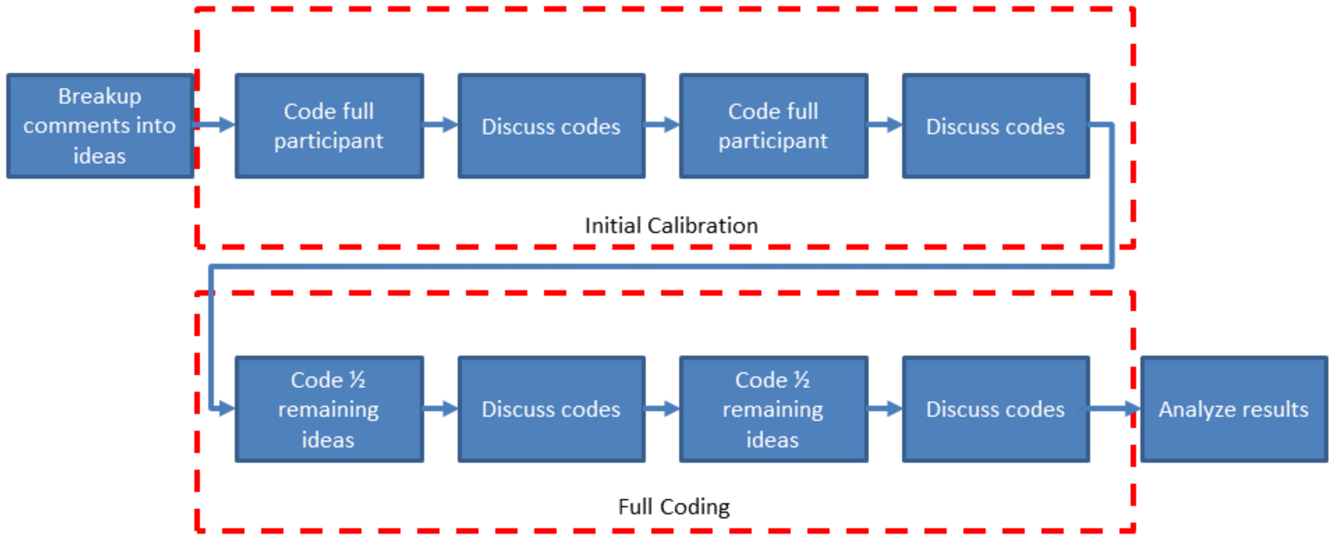

Figure 4: Summary of coding methodology

Coding Procedure Analysis

Before discussing the participants' codes from the coding process described above, certain limitations and characteristics of the process must first be addressed. Because the data was qualitative, researcher agreement, difficulties in coding, observations and the researcher background are relevant to all numerical results presented subsequently.

Interrater Reliability

Interrater reliability is a measure of agreement between multiple raters [28]. For this study, the raters were the two researchers that coded the ideas. The measure used to determine interrater reliability for this study was pre-discussion percent agreement. Because the discussion phase enforced a 100% percent agreement after discussion, post discussion agreement was not meaningful. Pre-discussion agreement was examined for each classification method separately and then together. Pre-discussion percent agreement is graphed in Figure 5. Initially, the percent agreement was very low. This was most likely because of the large number of codes, complexity of definitions, and a lack of a shared understanding between researchers. After the first discussion, agreement improved from an average of 5.9% to 57.4%. This is indicative of the effectiveness of an initial discussion when coding qualitative data. Subsequent discussions did not have this same effect. Instead, pre-discussion percent agreement fluctuated around 55% for the remainder of the coding. Variability between participant commenting styles may have been a limiting factor in pre-discussion agreement independent of the number of discussions that occurred previously.

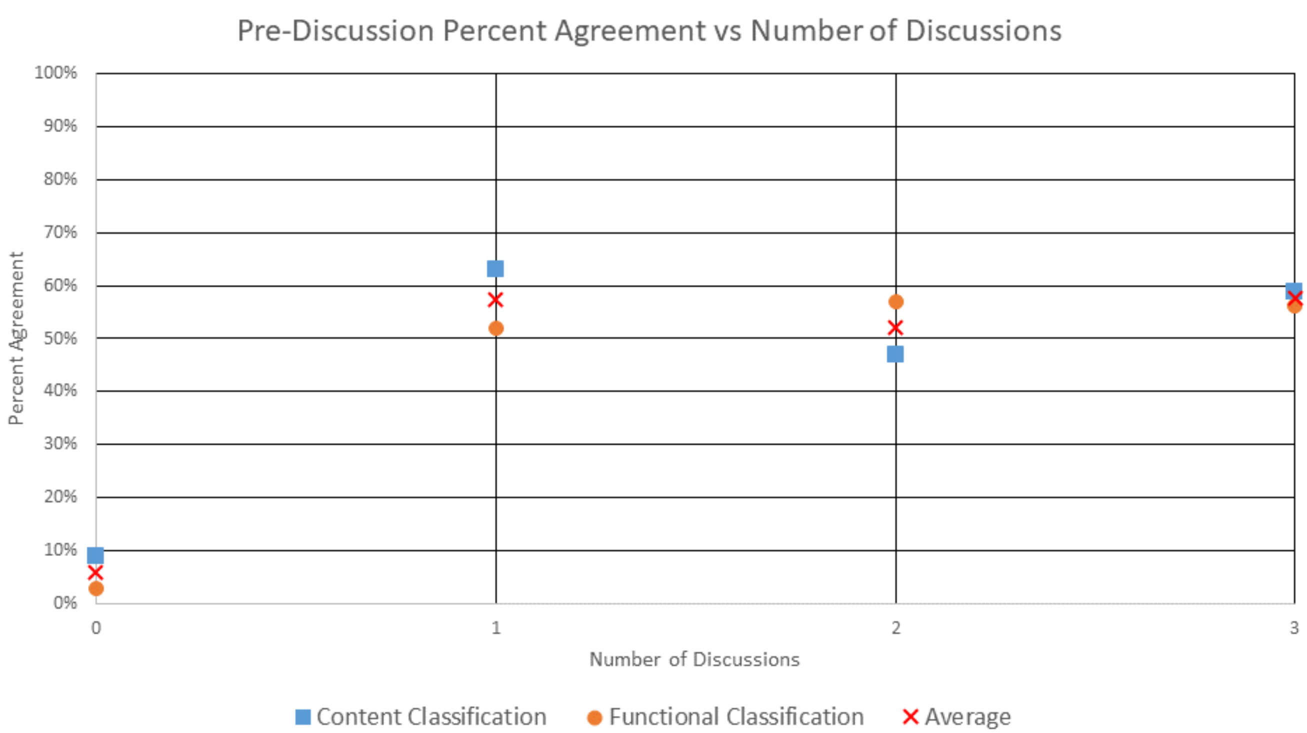

Figure 5: Pre-discussion agreement for content classification, functional classification, and their average

Pre-discussion agreement by participant also varied, with a standard deviation of 18.4% and 18.7% for content classification and functional classification respectively. This spread is illustrated in Figure 6 that shows the pre-discussion percent agreement by participant number.

The participant numbers are placed in the order that their ideas were coded, and the black boxes indicate the discussion groups. Despite the third discussion group containing six participants, this group contained approximately the same number of ideas as the final discussion group that had only two participants.

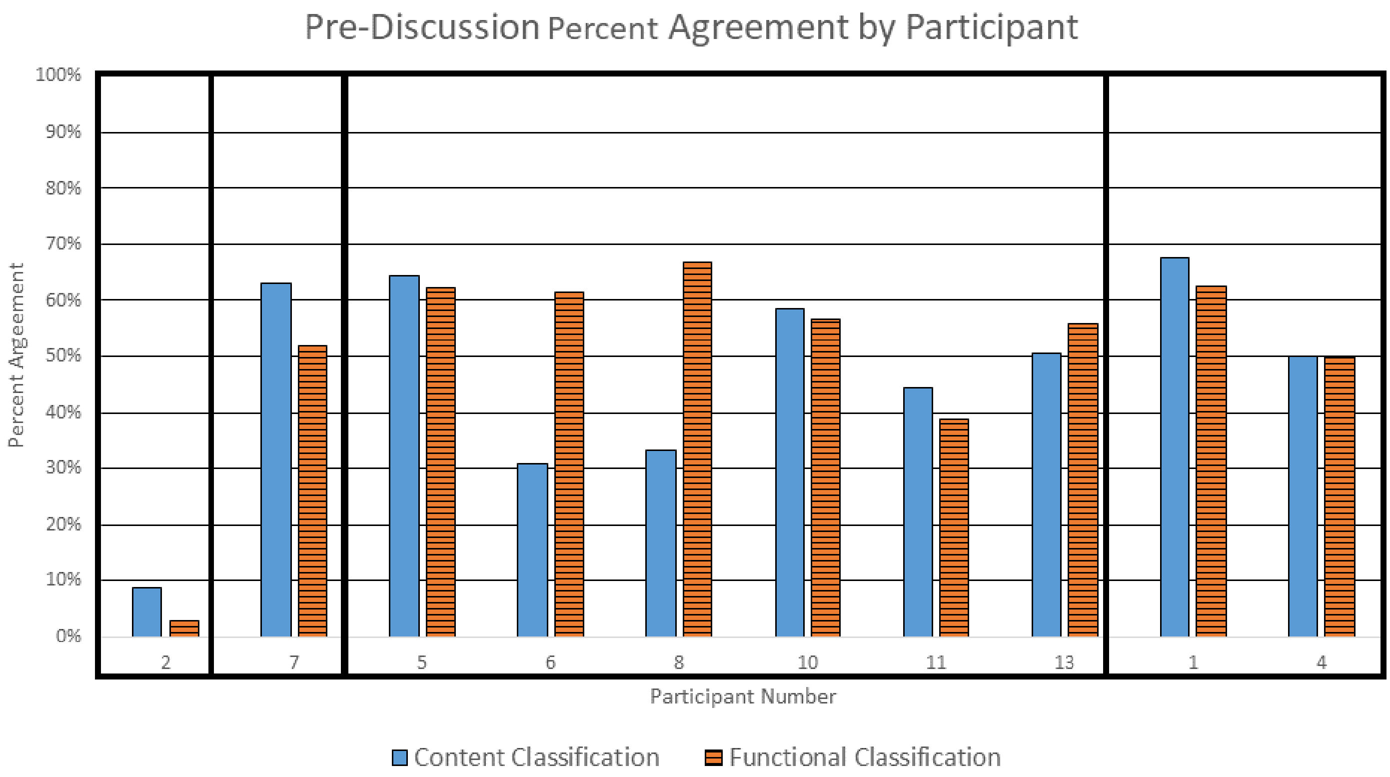

Figure 6: Content and functional classification pre-discussion percent agreement, organized by discussion groups

Finally, before coding began it was hypothesized that content classification pre-discussion agreement would be higher than functional classification agreement because the purpose of an idea is more abstract than the information. However, the average pre-discussion percent agreement for content classification and functional classification was 47.2% and 50.9% respectively. One potential reason to explain this difference is the background of the researchers. Both researchers have experience with leaving written feedback on technical communications and this experience may have influenced their perceptions of the purpose of an idea in a similar way.

Researcher Limitations and Observations

The primary difficulties faced by the researchers while working through the coding process are listed below:

1. As the coding process went on, the researchers' understanding of the code definitions grew. Despite this growth, participants who were coded first were not revisited.
2. The researchers had difficulty remembering decisions made about coding after spending time away from the coding process.
3. Researchers did not review the prompt directly before starting the coding process.
4. Occasionally, codes were chosen for an idea that did not agree with either researcher's original codes but codes that already had agreement between the researchers were not discussed.
5. Researchers had issues differentiating between Confirming and Assessing. This led to Assessing ideas primarily being negative.
6. Researchers had issues differentiating between Knowledge of Mistakes and both Knowledge of How to Proceed and Knowledge of Concepts.

The first observation made by the researchers regarding the lack of revisiting ideas was a limitation of the coding process. Because the discussions led to an increased understanding of the codes, revisiting codes may have led to a reclassification of some ideas. The goal of the initial discussions was to build the understanding of the codes quickly to minimize this effect. Figure 5 supports this decision, as the only discussion that had a large impact on pre-discussion agreement between researchers was the first discussion.

Both the second and third observation were considered in the coding process to help eliminate their impact. Detailed notes were kept by one of the researchers for each of the discussions and made available to both researchers while they independently coded the next set of ideas. This helped mitigate the need for each researcher to remember each decision made. As for the prompt, both researchers were familiar with the experiment and prompt before the coding process began despite not reviewing it directly. The researchers previously completed the same experiment discussed in the prompt both as students and UTAs in the FEP courses. One researcher also conducted the experiment as a GTA in the course sequence. Since the last time both researchers completed the experiment, the prompt had remained largely unchanged.

Observation four was an unexpected outcome of the coding process. Out of the 586 ideas that were coded by the researchers, only 40 content ideas and 20 functional ideas had codes that did not agree with either of the researchers' original codes. This is only 6.8% and 3.4% of the ideas that were coded, respectively, and this error should not significantly impact the code distributions.

Finally, the issues with differentiation between codes was expected because of the difficulty in creating a coding scheme that can capture all possible information in a unique way. The goal of synthesizing multiple coding methods was to minimize this effect. Another method to minimize this issue was forcing 100% agreement between researchers on each idea, making it less likely that a single researcher's confusion would propagate throughout the results.

Positionality Statement

My background as a researcher is relevant to the discussion of the coding results because of the qualitative nature of the data and the coding process. I am currently an undergraduate engineering student at Ohio State and took FEH during my first year. This is relevant because the writing sample used for this study was my own lab report I wrote while in FEH. Because of this, I was more familiar with the content of the report and the requirements of the assignment during the coding process. This familiarity may have impacted how I coded the feedback. Additionally, the participants were critiquing my own writing and while I did not actively consider this point, it may have unconsciously impacted my process.

After finishing the FEH courses I became a UTA for FE for 4 semesters, where I helped students conduct the experiment as a member of the teaching team. My experiences as a UTA informed my coding process and influenced how I viewed the participants' comments. While coding I thought about my own process I use while giving written feedback to students which may have influenced how I coded the ideas.

Both effects were minimized through the help of the second researcher who also coded the data. I am also several years removed from writing the report and was not an active UTA at the time of the coding. This removal may have also helped minimize the effects of my experiences.

Coding Results

Because the analysis of the written feedback was qualitative, a statistical analysis is not provided as it normally would be for quantitative results. Rather, results will be presented fully and discussed in terms of trends and observations rather than statistical outcomes. Additionally, the participants are often analyzed and compared in terms of their group. Two groups were chosen for this study: UTAs, and Experts. The Experts refer to the GTA and Faculty, Participant 10, 11 and 13, who have more expertise than the UTAs because of their field of study, years of experience, technical knowledge, and role.

Content Classification Results

The content classification code for an idea represented the explicit content of the idea. Table 12 shows the content codes and acronyms, definitions used by the researchers, and exemplar comments for each code. The acronyms listed above will be used in subsequent discussions of the content codes. KMC does not have an example because no comments received this code. Another code that was used sparingly was KCR which had only 2 ideas, each from a different participant. The percentage of total ideas for each code per participant, as well as the raw counts, are listed in Table 13. The large range of values is presented visually in Figure 7. Each participant had a unique distribution of ideas, with few discernable trends between participants. Most ideas for each participant were coded either as KC or KH and very few participants utilized KP or KR ideas.

Table 12: Content classification codes, definitions, and examples

Code	Definition	Examples
Knowledge of Performance (KP)	Score/Percentage correct	“(-1 Language & Precision)”
Knowledge of Response (KR)	Informs if response is correct or incorrect	“This is exactly right!”
Knowledge of the Correct Result (KCR)	Gives the correct response	“When I did the calculation with these values, I got 120 mph, not 60 mph”
Knowledge about Task Constraints (KTC)	Hints/explanations on type of task Hints/explanations on task-processing rules Hints/explanations on subtasks Hints/explanations on task requirements	“Missing several of the requirements for a lab report title page”
Knowledge about Concepts (KC)	Hints/explanations on technical terms Examples illustrating the concept Hints/explanations on the conceptual context Hints/explanations on concept attributes Attribute-isolation examples	"An error is something that cannot be controlled but is built into the experiment."
Knowledge about Mistakes (KM)	Number of mistakes Location of mistakes Hints/explanations on types of errors Hints/explanations on sources of errors	"The spacing in between lines here isn't quite even"
Knowledge about How to Proceed (KH)	Bug-related hints for error correction Hints/explanations on task-specific strategies Hints/explanations on task-processing steps Guiding questions Worked-out examples	“Suggest making this sentence the first sentence of the paragraph”
Knowledge about Metacognition (KMC)	Hints/explanations on metacognitive strategies Metacognitive guiding questions	N/A

Table 13: Content code percentages and counts for all participants

Participant	Roles	Course	KPs	KRs	KCRs	KTCs	KCs	KMs	KHs	KMCs
1 (n=235)	UTA	1281	0.0% (n=0)	6.8% (n=16)	0.4% (n=1)	3.4% (n=8)	34.0% (n=80)	9.4% (n=22)	46.0% (n=108)	0.0% (n=0)
2 (n=34)	UTA	1281	17.6% (n=6)	0.0% (n=0)	0.0% (n=0)	8.8% (n=3)	20.6% (n=7)	8.8% (n=3)	44.1% (n=15)	0.0% (n=0)
4 (n=4)	UTA	1181	0.0% (n=0)	0.0% (n=0)	0.0% (n=0)	0.0% (n=0)	100.0% (n=4)	0.0% (n=0)	0.0% (n=0)	0.0% (n=0)
5 (n=45)	UTA	1181	0.0% (n=0)	8.9% (n=4)	0.0% (n=0)	6.7% (n=3)	26.7% (n=12)	6.7% (n=3)	51.1% (n=23)	0.0% (n=0)
6 (n=13)	UTA	1181	0.0% (n=0)	0.0% (n=0)	0.0% (n=0)	23.1% (n=3)	38.5% (n=5)	23.1% (n=3)	15.4% (n=2)	0.0% (n=0)
7 (n=27)	UTA	1181	0.0% (n=0)	18.5% (n=5)	0.0% (n=0)	3.7% (n=1)	44.4% (n=12)	3.7% (n=1)	29.6% (n=8)	0.0% (n=0)
8 (n=15)	UTA	1181	0.0% (n=0)	26.7% (n=4)	0.0% (n=0)	6.7% (n=1)	26.7% (n=4)	6.7% (n=1)	33.3% (n=5)	0.0% (n=0)
10 (n=118)	GTA	1281	0.0% (n=0)	6.8% (n=8)	0.8% (n=1)	2.5% (n=3)	50.8% (n=60)	7.6% (n=9)	31.4% (n=37)	0.0% (n=0)
11 (n=18)	Faculty	1181	0.0% (n=0)	11.1% (n=2)	0.0% (n=0)	0.0% (n=0)	50.0% (n=9)	0.0% (n=0)	38.9% (n=7)	0.0% (n=0)
13 (n=77)	ETC	-	0.0% (n=0)	2.6% (n=2)	0.0% (n=0)	0.0% (n=0)	42.9% (n=33)	1.3% (n=1)	53.2% (n=41)	0.0% (n=0)

Box and whisker plots were used to visualize the distribution of the percentage of ideas for category by group. Figure 8 shows all 8 categories, with the average of each category per group labeled with an “X”, and data points labeled with an “O” or with one of the lines on the box and whisker plots. The categories with the most similar distributions were KP, KCR, and KMC. This is primarily because very few participants used these types of ideas in their feedback. KR ideas were also used in a similar way between the UTA and Expert groups. Their median and average percentage of ideas were nearly identical, but the UTA group had a larger spread. Here, spread refers to the difference between the minimum and maximum proportion in the group. This larger spread of data for the UTA group was true for 6 out of the 8 categories where the UTAs had an average spread of 27.7% while the Experts had an average spread of 6.2%. UTAs and Experts also had a similar distribution for KM ideas where all but one UTA kept their utilization under 10%.

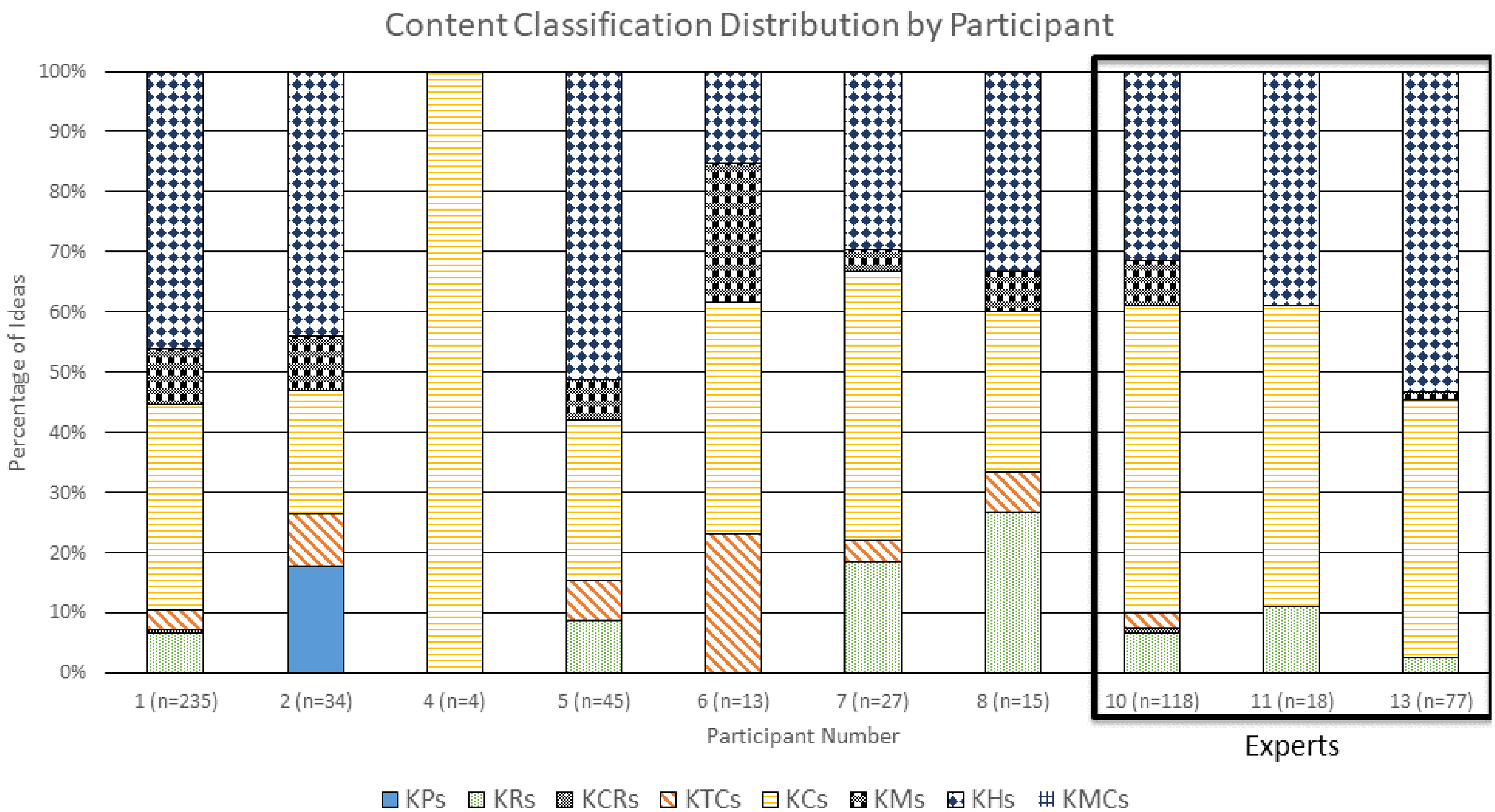

Figure 7: Distribution of content classified ideas by participant

Even though KTC ideas accounted for a small percentage per group like KP, KCR, and KMC, the distribution of ideas differs between groups. The average percentage of ideas for the UTAs was 7.5% while the average for the Experts was 0.8%. The UTAs also had a higher spread than the Experts (23% and 2.5% respectively).

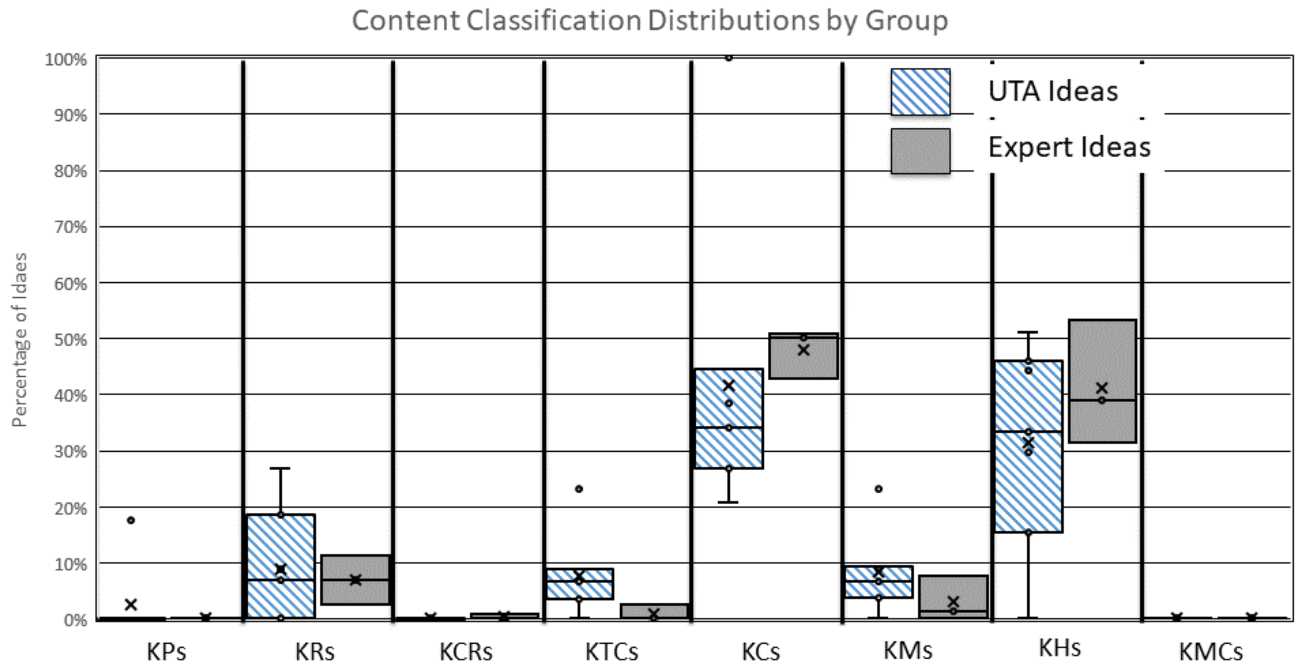

Figure 8: Percentage of total ideas for each content code by group

As previously noted, KC and KH accounted for the largest percentage of ideas for all participants, so they also accounted for the largest percentage of ideas for both UTA and Expert groups. Despite accounting for the greatest proportion of ideas for both groups, the distributions between the two groups for each category are not alike. KC and KH are the only two categories that the Expert group used at a higher rate than the UTA group. Experts had an average of 47.9% and 41.2% for proportion of KC and KH ideas, while the UTAs had an average of 41.6% and 31.4% for proportion of KC and KH ideas. The Experts had a much smaller spread than the UTAs in these two categories as well. This is primarily because one participant only used KC ideas (which then leads to a 0% use of KH ideas). For KC, the UTA spread was 79.4% and the Expert spread was 8%, a difference in spread of 71.4%. The difference in KH spread was less pronounced: the UTAs had a spread of 51.1% while the Experts had a spread of 21.9%.

In general, while some category distributions were similar between the UTA and Expert groups, most of the ideas used by the groups were different. The categories that had the most agreement between the groups accounted for the smallest number of ideas per participant. Categories with strong differences between the distributions accounted for most of the ideas left by each participant.

Functional Classification Results

Functional codes were codes related to the purpose of an idea. The final code definitions used by the researchers and exemplars of each type of idea are in Table 14. The full percentage of ideas each code type accounted for by participant, along with the full counts, are presented in Table 15. Only 12 ideas were coded as Motivating ideas while 361 ideas were coded with the Correcting code. These represent the two extremes for the idea counts. The distribution of percentages per individual is highlighted in Figure 9. Like the content classification results, most similarities between participants came from the ideas used the least by each group. Confirming, Informing, and Motivating ideas were either not used or used very sparingly by the participants. For the remaining codes, the only clear trend was Correcting ideas accounting for a large proportion of the total ideas left by every participant. The use of Assessing, Indicating, and Instructing ideas varied heavily between individuals.

Table 14: Functional classification codes, definitions, and examples

Code	Definition	Examples
Confirming	Tells students that their understanding of the material is correct	“Good explanation of data analysis”
Assessing	Evaluates student performance	"The biggest opportunities for improvement have to do with how you are describing and processing the results."
Informing	Add information that the students may lack	“In the experimental methodology section, you should just be explaining what you did. You don’t need to justify every decision you made.”
Indicating	Show area of interest for student to re-examine themselves	“See comments about inclusion of quotes”
Correcting/Regulating	Provide information meant to replace inaccurate student knowledge or misconception	“random/human error is not a valid source of error”
Motivating	Encourages/facilitates student desire to learn/continue effort	“Think about how you could improve the wording to make this sentence stronger.”
Instructing	Tweak a mostly correct understanding, differentiate between similar concepts, specifying conditions for rules, provide supplemental information meant to enhance understanding further, etc.	“In general, the report contains all the elements, but can be further improved through making writing more concise and organizing the data being presented and discussed in a more straightforward manner”

Table 15: Functional classification percentage of ideas per participant

Participant	Roles	Course	Confirming	Assessing	Informing	Indicating	Correcting	Motivating	Instructing
1 (n=235)	UTA	1281	2.6% (n=6)	14.9% (n=35)	6.4% (n=15)	4.3% (n=10)	67.2% (n=158)	2.6% (n=6)	2.1% (n=5)
2 (n=34)	UTA	1281	0.0% (n=0)	26.5% (n=9)	0.0% (n=0)	0.0% (n=0)	64.7% (n=22)	0.0% (n=0)	8.8% (n=3)
4 (n=4)	UTA	1181	0.0% (n=0)	50.0% (n=2)	0.0% (n=0)	0.0% (n=0)	50.0% (n=2)	0.0% (n=0)	0.0% (n=0)
5 (n=45)	UTA	1181	8.9% (n=4)	8.9% (n=4)	0.0% (n=0)	2.2% (n=1)	55.6% (n=25)	0.0% (n=0)	24.4% (n=11)
6 (n=13)	UTA	1181	0.0% (n=0)	0.0% (n=0)	0.0% (n=0)	23.1% (n=3)	76.9% (n=10)	0.0% (n=0)	0.0% (n=0)
7 (n=27)	UTA	1181	18.5% (n=5)	3.7% (n=1)	14.8% (n=4)	0.0% (n=0)	51.9% (n=14)	0.0% (n=0)	11.1% (n=3)
8 (n=15)	UTA	1181	13.3% (n=2)	26.7% (n=4)	6.7% (n=1)	0.0% (n=0)	53.3% (n=8)	0.0% (n=0)	0.0% (n=0)
10 (n=118)	GTA	1281	3.4% (n=4)	14.4% (n=17)	5.1% (n=6)	17.8% (n=21)	49.2% (n=58)	4.2% (n=5)	5.9% (n=7)
11 (n=18)	Faculty	1181	11.1% (n=2)	5.6% (n=1)	0.0% (n=0)	22.2% (n=4)	27.8% (n=5)	0.0% (n=0)	33.3% (n=6)
13 (n=77)	ETC	-	2.6% (n=2)	6.5% (n=5)	1.3% (n=1)	5.2% (n=4)	76.6% (n=59)	1.3% (n=1)	6.5% (n=5)

Like the content classification, the participants were grouped by role and examined using box and whisker plots to show the distribution of the percentage of ideas for each category. These charts are shown in Figure 10. Just as in the breakdown by participant, the similar distributions between groups were Confirming, Informing and Motivating. These were also the three codes used by the Experts the least and 3 of the 4 least used codes by the UTAs. The difference in spread between the Experts and UTAs for these codes was less than 10% and the difference in averages between the two groups was less than 2% in all three cases.

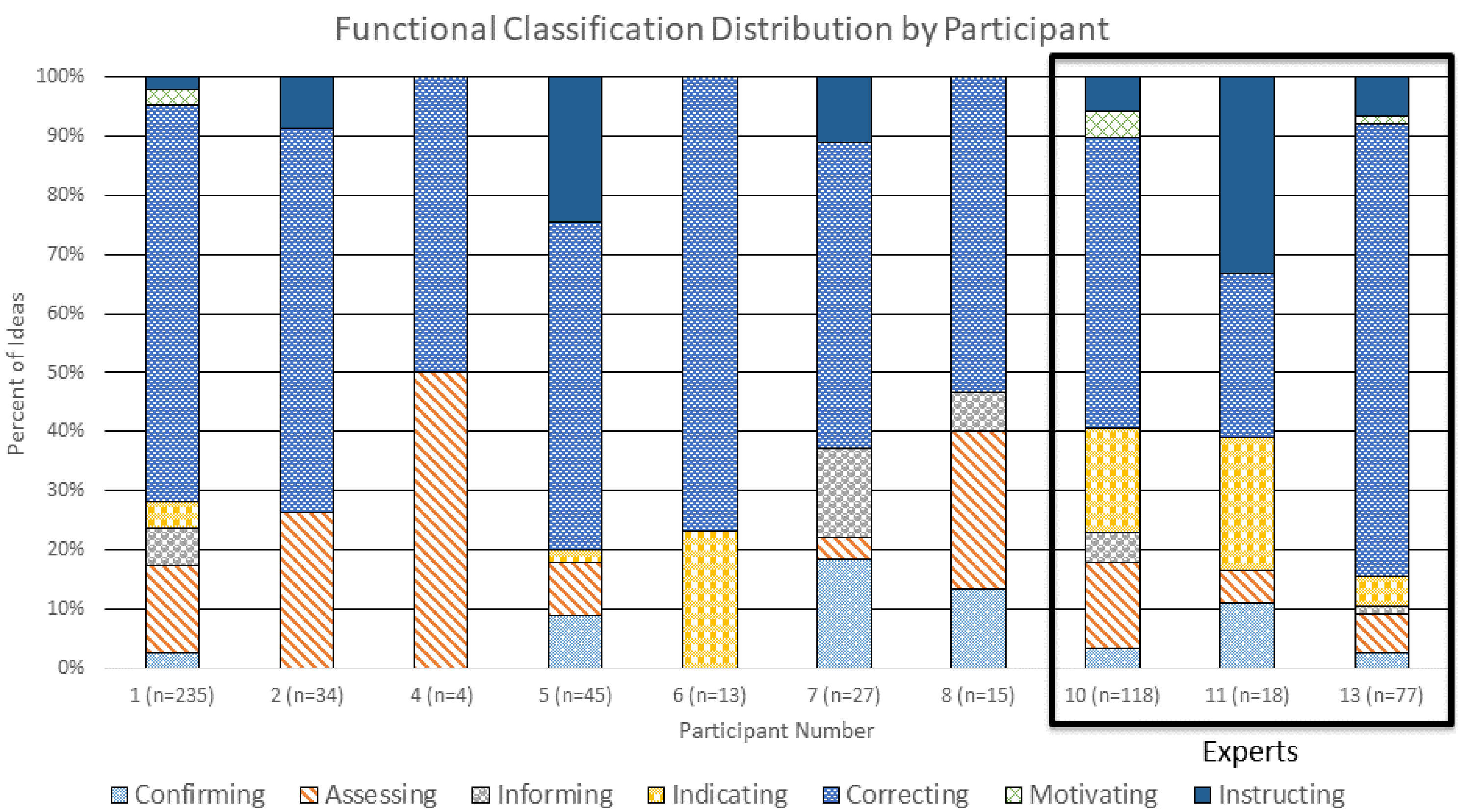

Figure 9: Functional classification percentage of ideas per category per individual

The distributions for the remaining codes were different between the UTA and Expert groups. Table 16 shows this difference numerically by highlighting average percentage of ideas and maximum spread for each remaining category by group. The largest difference in spread came from the Assessing code. Assessing also had the second largest difference in average percent of ideas. The code with the largest difference in average percent of ideas was Indicating, although all differences in averages between the groups were similar at approximately 9.5%. Out of these four codes, the use of Instructing ideas was the most similar between groups. Instructing had the smallest difference for both spread and average percentage. UTAs and Experts had similar average usage of Correcting ideas, which accounted for the largest percentage of ideas for both groups. Despite this similarity there was a large difference in spread between the two groups, which was the second largest difference in spread for all functional codes.

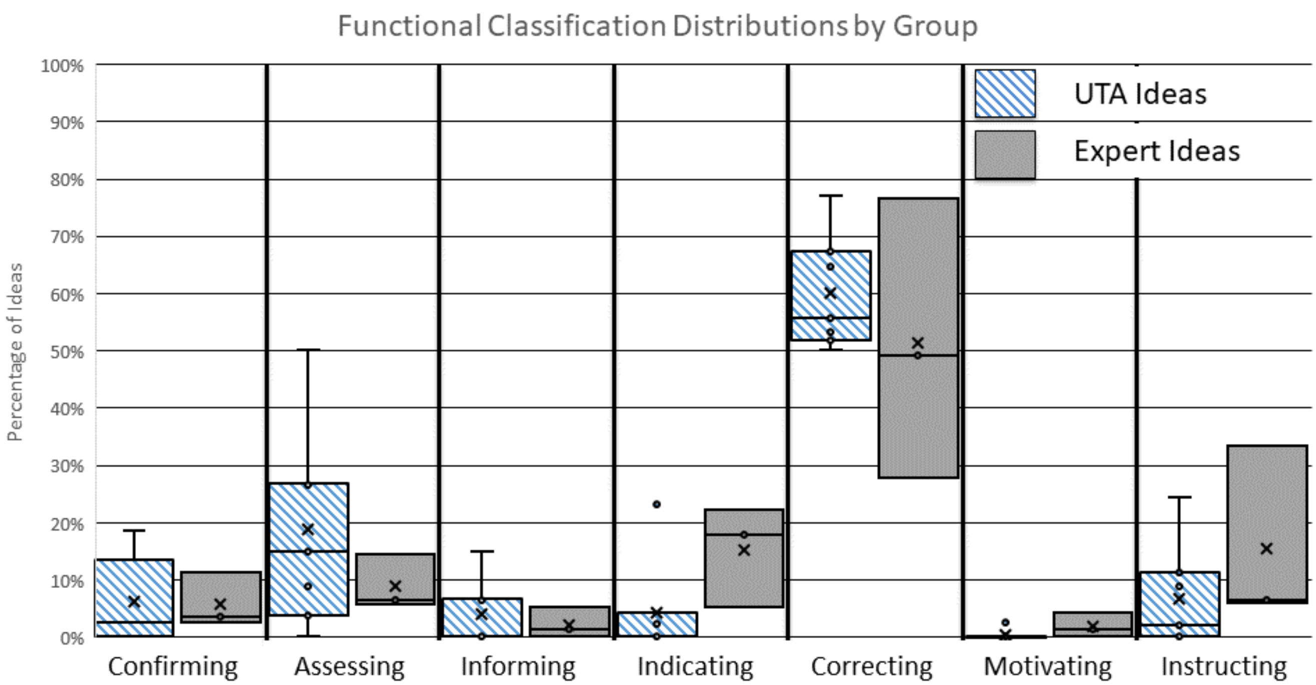

Figure 10: Functional classification distributions by group

Table 16: Numerical analysis of codes where UTAs and Experts differed

Code	Statistic	UTA	Expert	Difference
Assessing	Spread	50.00%	8.85%	**41.15%**
	Average	18.7%	8.8%	**9.84%**
Indicating	Spread	23.08%	17.03%	**6.05%**
	Average	4.2%	15.1%	**-10.85%**
Correcting	Spread	26.92%	48.85%	**-21.92%**
	Average	59.9%	51.2%	**8.76%**
Instructing	Spread	24.44%	27.40%	**-2.96%**
	Average	6.6%	15.3%	**-8.61%**

Like the content classification, the UTA and Expert groups had strong agreement for the codes that were used the least. The three codes that were similar between the UTAs and Experts accounted for an average of 10.53% and 9.67% of ideas respectively. This means that the codes that were different between the groups accounted for around 90% of ideas generated by the two groups. So, the Expert and UTA groups were similar in the codes they used sparingly but differed in the codes used most of the time.

Code Discussion

One goal of this analysis was to compare the UTA and Expert group feedback. Because the training is created by a group like our Expert group, these comparisons can give insight into how training may have influenced the UTAs. As previously noted, it is important to recognize the limitations of the small sample size and analysis techniques used when assessing the results.

Content Classification Discussion

The three codes with the most similarity between the UTA and Expert groups (KP, KCR, and KMC) were also among three groups with the lowest percentage of ideas for the Expert and UTA groups. This relationship indicates that the UTA and Expert groups had similar preferences for what type of content to NOT leave as feedback. Relating back to the definition for feedback as shown in Equation 1, KP and KCR only handle the verification portion of feedback. Verification is fundamentally simpler than elaboration because there is a limit on the ways verification can be expressed that does not exist for elaboration. For verification, either the item is correct or incorrect, and the student is either informed or not informed of the correct response. By carrying less information compared to elaborated feedback, the Experts and UTAs may view verification only feedback as less effective or helpful for students. This idea may be amplified by the Experts because examples of poor feedback in training may be both easier for the Experts to generate and/or easier for the UTAs to remember. This does not necessarily mean the KP and KCR feedback is less effective than other content left as feedback, but only that there may be a shared idea between the UTAs and Experts that it is poor feedback for technical writing.

Because KMC is a more complex form of feedback than simply giving a score or sharing the correct response, KMC may have been excluded for different reasons than KP or KCR. For example, the UTAs and Experts may have thought that KMC ideas would be effective but believed the marginal benefits over other types of feedback was not worth the additional effort needed to generate such ideas. Another possibility is that both UTAs and Experts were unaware of how to generate KMC ideas because of a lack of information or training. This is unlikely for KP or KCR because of their relative simplicity compared to KMC ideas. Finally, both groups may not have seen any opportunity to utilize this code within the writing sample.

The KTC distribution differences between the UTAs and Expert groups are also of interest. Because the UTAs in the FEP courses are responsible for scoring and giving feedback on most student work, they are often very familiar with the requirements for each assignment. This is especially true for the assignment that was chosen because the UTAs would have also had to complete a similar assignment when they took the course as it had been largely static over the past few years. Only two of the three Experts would have recently helped conduct the experiment because the ETC faculty member was not a member of the instructional team for the FEP courses. While the remaining two Experts would be familiar with the assignment and the requirements, they are not responsible for scoring students on their ability to meet the requirements like the UTAs are. The higher average use of KTC ideas by the UTAs is then explained by the UTAs' increased familiarity with assignment requirements compared to the Expert groups.

The remaining categories (KR, KC, KM and KH) were the categories with the largest differences between the UTA and Expert groups. Differences between the groups represent that the groups may see the effectiveness of certain content within feedback differently. Since the Experts are responsible for generating the training that was completed by the UTAs, this may also indicate that the training does not effectively communicate the Experts' ideas of ideal content to use in feedback to the UTAs or that UTAs rely on other experiences to inform their feedback. This does not mean that the Experts' idea of ideal content for student feedback is the most effective method.

Finally, the spread of all content distributions for the UTA group is indicative of the similarities and differences between the individual UTAs. Besides KCR and KMC, which will be disregarded because they were either not used or used extremely sparingly, the minimum spread of the percentage of ideas for a category was 17.7% and the maximum was 79.4% with an average spread of 36.8%. This large spread between the minimum and maximum use of each kind of content code may be related to differences in training between individuals, issues with training covering effective content for feedback on technical communication, or the differences may be indicating that other experiences besides training influenced the content related feedback methods used by the UTAs.

Functional Classification Discussion

The functional codes that were generated are tied directly to the purpose or function of the idea that is being examined. As with the content classification, the three most similar distributions between the UTAs and Experts (Confirming, Informing and Motivating) were also some of the least utilized types of ideas by both groups. The similarity in averages and spread for all three categories, and the relatively low value of each average, indicates that both the UTAs and Experts did not see the need to use these types of comments on this writing sample.

One aspect not yet discussed that may have influenced the trends in code distributions are past students' reactions, or lack of reactions, to feedback left by the participants. If participants have had experiences with students who do not act on feedback they receive, the participants may have been more inclined to leave feedback that directly fixes an issue. The codes that do not correspond to a direct change were exactly the codes that were used the least by both groups. Because both UTAs and Experts have previous experience giving feedback on technical writing to varying extents, a lack of previous student review may be a shared experience among them.

Out of the remaining categories, the UTAs had a lower average percent use than the Experts with only 2 codes: Indicating and Instructing. Indicating ideas were primarily ideas that directed the reader to view other comments, while Instructing ideas focused on adjusting the student's understanding from nearly correct to correct. Experts focusing more on Indicating ideas may reflect on the style of commenting of the Experts compared to the UTAs. Instead of directing students to similar comments that were already left, UTAs may have instead just repeated the comment without any mention of a similar comment appearing elsewhere. These two methods effectively communicate the same information to the students but place more responsibility on the students to re-examine previous areas of the assignment. As for the Instructing ideas, because these ideas rely on recognizing that the student is nearly correct and then being able to generate an idea that would give the student the correct adjustment, a greater amount of expertise may be needed to use them. In general, the Experts have more expertise with high level technical writing than the UTAs because of their attainment of advanced degrees, and greater years of teaching experience. So, the Experts having a larger average use of Instructing comments compared to the UTAs can be explained by their increased expertise.

Correcting ideas were the most used ideas for nearly all participants. While this demonstrates that all participants corrected mistakes more than all other functions of feedback, the proportion of ideas with this code still varied. Within the Experts, the spread between participants for the Correcting code is much higher than the spread between Expert participants for any other functional code, and the spread in Correcting for the UTA participants. The two next largest spreads between participants for the Experts occurred with Instructing and Indicating. After this there was a large jump to the fourth largest spread. Indicating, Instructing, and Correcting ideas could have been potentially interchanged depending on the error the participant was examining. This relationship may help explain why the spread for these three codes are much larger than the other codes. One reason this same trend was not present in the UTA group may be from differing experiences between the UTAs and Experts that are used to inform feedback styles. Another reason the Expert spread may have been greater is that the Experts do not all share common experiences to draw from regarding how to handle errors when giving feedback.

Like the content classification, the difference in how both groups handled most of their ideas is indicative of how effectively training information generated by those in the Expert group is communicated to the UTAs. It can also relate to other experiences that inform UTA feedback methods, and a difference in experience may be responsible for differences between UTA and Expert ideas. Both effects may also manifest in the spread between participants within the UTA group for each functional category. Each UTA participant may have had different previous experiences, such as training or external experiences, that shaped their written feedback methods to be different from their peer participants.

Future Work

The current presented study primarily focused on trends seen from UTA and Expert content and functional distributions. Because of this limited approach there are many paths that have not yet been explored with the current data. One approach will be to return to the code counts and examine correlations between content and functional classifications to see if any content codes align with any functional codes or vice versa. This will help build a stronger understanding of the synthesized coding schemes. Another analysis of interest is examining the functional and content code structure of full comments. This analysis will examine the codes that make up a comment and in the order they occur in, while marking specific structures of interest. Structures of interest will be those that appear frequently among participants, or any structures unique to a particular group.

Separate from this data collection, a focus group was also conducted with UTA participants after they had completed the coding segment outlined above. This focus group was centered around the experiences that informed the participants written feedback methods. Analysis of the completed focus group will be part of a future study. The focus group was conducted before the analysis of the coded data was complete, so a follow up focus group may be necessary to ask questions related to the coding results.

Finally, the study may be expanded in the future to examine the trends highlighted here in further detail. This would involve a new data set with a much larger sample size from all participant groups, as well as an expanded coding methodology that would address the limitations in the current procedure. The goal for this expanded study would be to demonstrate these trends on a larger scale, and potentially show statistical significance to best assess the current training procedure as it relates to written feedback.

Conclusion

This study sought to be the first step in examining the training related to UTA written feedback within the FEP courses. Currently, training consists primarily of scoring assignments and familiarizing UTAs with the material that is needed to complete each assignment. Written feedback is left to examples during a single day of training that occurs once a year. Using this lack of coverage as the research motivation, a small group of FEP teaching team members and a member of the ETC faculty scored and provided feedback on a student writing sample. These comments were broken down into single ideas and coded using two separate coding schemes, each synthesized from the literature. Analyzing these codes showed agreement between the UTA and Expert groups in what content they did NOT include, and which function of ideas were avoided, however there was disagreement between these groups in the categories that were most used by all participants. The differences between the UTA and Expert functional and content classification distributions shows that there is not a shared idea between groups as to what information comments should contain and how the comments should express the information. Training may not effectively communicate the type of feedback Experts would use, or UTAs may be drawing on other experiences to inform their feedback methods. Future work will explore the experiences that inform the UTA feedback process through a focus group with the UTA participants that completed the coding segment of this study. In the future, this study may be expanded upon to increase the participant count and adjust the coding methodology to improve upon the observations made here.

References

[1] S. Last, Technical Writing Essentials, Victoria: University of Victoria, 2019.

[2] S. Pneena and C. Romanowski, "A Message from Recent Engineering Graduates in the Workplace: Results of a Survey on Technical Communication Skills," *Journal of Engineering Education,* vol. 90, no. 4, pp. 685-693, 2001.

[3] C. Nicometo, K. Anderson, T. Nathans-Kelly, S. Courter and T. McGlamery, "“More Than Just Engineers”: How Practicing Engineers Define and Value Communication Skills On the Job," in *American Society for Engineering Education*, Louisville, 2010.

[4] M. Riemer, "Communication Skills for the 21st Century Engineer," *Global Journal of Engineering Education,* vol. 11, no. 1, pp. 89-100, 2007.

[5] ABET: Engineering Accreditation Commission, "Criteria for Accrediting Engineering Programs," ABET, Baltimore, 2019.

[6] J. Hattie and H. Timperley, "The Power of Feedback," *Review of Educational Research,* vol. 77, no. 1, pp. 81-112, 2007.

[7] V. Shute, "Focus on Formative Feedback," Educational Testing Services, Princeton, 2007.

[8] D. Nicol and D. Macfarlane-Dick, "Formative assessment and self-regulated learning: a model and seven principles of good feedback practice," *Studies in Higher Education,* vol. 31, no. 2, pp. 199-218, 2006.

[9] K. Rodgers, A. Horvath, H. Jung, A. Fry, Diefes-Dux and Heidi, "Students' Perceptions of and Responses to Teaching Assistant and Peer Feedback," *Interdisciplinary Journal of Problem-Based Learning*, vol. 9, no. 2, 2014.

[10] M. Rucker and S. Thomson, "Assessing Student Learning Outcomes: An Investigation of the Relationship Among Feedback Measures," *College Student Journal*, vol. 37, no. 3, 2003.

[11] The Ohio State University Department of Engineering Education, "First-Year Engineering Program," [Online]. Available: https://eed.osu.edu/academics/first-year-engineering-program.

[12] The Ohio State University Department of Engineering Education, "Eligibility and Course Structure," The Ohio State University Department of Engineering Education, [Online]. Available: https://eed.osu.edu/academics/undergraduate/first-year/fundamentals-engineering.

[13] The Ohio State University Department of Engineering Education, "EED Undergraduate Teaching Assistant Policies," The Ohio State University Department of Engineering Education, 2018. [Online]. Available: https://eed.osu.edu/sites/default/files/uploads/eed_uta_policies_au18_v2.pdf.

[14] K. Kecskemety, A. Theiss and R. Kajfez, "Enhancing TA Grading of Technical Writing: A Look Back to Better Understand," in *American Society for Engineering Education*, Seattle, 2015.

[15] K. A. Neuendorf, The Content Analysis Guidebook, Thousand Oaks: SAGE Publications, 2017.

[16] R. Kulhavy and W. Stock, "Feedback in Written Instruction: The Place of Response Certitude," *Educational Psychology Review*, vol. 1, no. 4, pp. 279-308, 1989.

[17] B. Mason and R. Bruning, "Providing Feedback in Computer-based Instruction: What the Research Tells Us," University of Nebraska - Lincoln, Lincoln, 2001.

[18] S. Narciss, "Feedback Strategies for Interactive Learning Tasks," in *Handbook of Research on Educational Communications and Technology*, New York, Lawrence Erlbaum Associates, 2008, pp. 125-144.

[19] D. Butler and P. Winne, "Feedback and Self-Regulated Learning: A Theoretical Synthesis," *Review of Educational Research,* vol. 65, no. 3, pp. 245-281, 1995.

[20] L. Cusella, "Feedback, Motivation, and Performance," in *Handbook of Organizational Communication*, Newbury Park, SAGE Publications, 1987, pp. 624-678.

[21] G. Sales, "Adapted and Adaptive Feedback in Technology-Based Instruction," in *Interactive Instruction and Feedback*, Englewood Cliffs, Educational Technology Publications, 1993, pp. 159-175.

[22] W. Wager and E. Mory, "The Role of Questions in Learning," in *Interactive Instruction and Feedback*, Englewood Cliiffs, Educational Technology Publications, 1993, pp. 55-73.

[23] D. Gilman, "Comparison of several feedback methods for correcting errors by computer-assisted instruction," *Journal of Educational Psychology*, vol. 60, no. 6, pp. 503-508, 1969.

[24] J. Merril, "Levels of Questioning and Forms of Feedback: Instructional Factors in Courseware Design," *Journal of Computer Based Instruction*, vol. 14, no. 1, pp. 18-22, 1987.

[25] R. Overbaugh, "Research-based guidelines for computer-based instruction development," *Journal of Research on Computing in Education*, vol. 27, no. 1, 1994.

[26] S. Pressey, "Development and Appraisal of Devices Providing Immediate Automatic Scoring of Objective Tests and Concomitant Self-instruction," *Journal of Psychology*, vol. 30, pp. 417-447, 1950.

[27] B. Schimmel, "Providing Meaningful Feedback in Courseware," in *Instructional Designs for Microcomputer Courseware*, Hillsdale, Lawrence Erlbaum Associates, 1988, pp. 183-196.

[28] J. Kreutzer, J. DeLuca and B. Caplan, Encylcopedia of Clinical Neuropsychology, Springer: Springer-Verlag, 2011.